

Test von Steuergeräten mit LIN 2.0 Interface

*Michael Schmidt, Manfred Schneider
GÖPEL electronic GmbH, Jena*

Kurzfassung

Mit Einführung des gegenüber seiner Vorgängerversion wesentlich komplexeren LIN Standards 2.0 in der Automobilelektronik sind nicht nur die Prüfanforderungen, sondern auch die Möglichkeiten der Prüfung von Steuergeräten mit LIN Interface deutlich gewachsen. So können beispielsweise die im LIN-Standard 2.0 definierten Diagnoseumfänge KWP 2000 für den Test der Steuergeräte ausgenutzt werden. Um derartige Tests effektiv durchführen und vor allem programmieren zu können, sind intelligente Schnittstellenbaugruppen erforderlich, die vom Hostsystem über einfache, parametrierbare Funktionsaufrufe bedient werden und selbstständig komplexe Kommunikationsabläufe auf dem LIN-Bus generieren bzw. auswerten.

Der vorliegende Beitrag beschreibt eine Familie von LIN 2.0 Interface Modulen und Ihre Einbindung in National Instruments Standardumgebungen wie PXI, TestStand und LabVIEW.

Einleitung

Der LIN-Bus wurde Ende der Neunziger Jahre als Standard definiert und hat seitdem eine sehr rasante Verbreitung in der Automobilelektronik erzielt. Ausgangspunkt seiner Einführung war die vorhersehbare Notwendigkeit, den Kommunikationsumfang der aufgrund hoher Anzahl von Steuergeräten und hoher Datenmengen im Fahrzeug an die Auslastungsgrenze geratenen CAN-Netzwerke zu verringern. Aus ökonomischer Sicht galt es, dem relativ aufwendigen CAN-Interface eine zu günstigerem Preis realisierbare Alternative gegenüber zu stellen.

Ein LIN-Netzwerk verbindet im Fahrzeug stets eine begrenzte Anzahl von Steuergeräten, die eine in sich geschlossene Funktionseinheit bilden. Die Ankopplung einzelner LIN-Subsysteme an das zentrale CAN-Netz des Fahrzeuges erfolgt jeweils über ein Steuergerät mit Gateway-Funktion. Die Kommunikation auf dem LIN-Bus wird von einem als Master fungierenden Busteilnehmer kontrolliert, die übrigen Steuergeräte arbeiten im Slave Mode.

Eine in ein universelles Funktionstestsystem zu integrierende LIN-Ansteuerbaugruppe muss in der Lage sein, alle möglichen Betriebsarten gemäß LIN-Spezifikation 2.0 zu unterstützen. Neben der eigentlichen Steuergerätekommunikation gehören dazu auch Diagnosedienste, deren Umfang in einem sogenannten Keyword-Protokoll (KWP 2000) definiert ist. Zur Einhaltung realer Zeitregime in der Kommunikation ist die On-Board Generierung und Auswertung von Botschaftsfolgen erforderlich. Dies erfordert in der Firmware der LIN-Controller zu hinterlegende Funktionen, die durch externe Aufrufe bzw. Parameterübergabe aktiviert werden können.

Die genannten Aufgaben stellen Schwerpunkte in der Firmwareentwicklung von LIN-Interfaces dar, welche in den folgenden Abschnitten noch eingehender beschrieben werden. Nicht weniger Bedeutung ist der Systemintegration beizumessen, welche einerseits die Bereitstellung einer plattform-unabhängigen Treiberunterstützung, andererseits die

Anbindung standardisierter Datenbasen im ldf-Format (LIN Description Files) auf Bediener Ebene beinhaltet.

Hardwareseitig liegen die Herausforderungen neben der Unterstützung eines gängigen Spektrums von Hardwareplattformen vor allem in der Synchronisierung der LIN-Kommunikation mit konventionellen Messverfahren. Um beispielsweise die Reaktionszeit zu messen, die ein Steuergerät für die Umsetzung eines bestimmten LIN-Befehls benötigt, ist es notwendig, auf einzelne Botschaften triggern zu können.



Bild 1: Komplexes Funktionstestsystem für Kfz-Steuergeräte auf PXI-Basis mit integrierter LIN 2.0 Schnittstelle

Hardwareplattformen

Analysiert man die für Testaufgaben typischerweise eingesetzte Messtechnik von der Entwicklung elektronischer Baugruppen über die Fertigung bis hin zum After-Sales-Service, so gibt es in einzelnen Abschnitten des Produktlebenszyklus unterschiedliche Verbreitungen einzelner Plattformen. Generell ist festzustellen, dass sich der PXI-Standard seit seiner

Einführung durch National Instruments Ende der Neunziger Jahre vehement durchgesetzt hat. Er bildet aktuell die verbreitetste Basis komplexer Funktionstestsysteme in der Fertigung elektronischer Baugruppen und hat bis dato gängige Standards wie VXI, VME oder GPIB-Lösungen praktisch zu einem Nischendasein verbannt. Da PXI-Systeme modular aufgebaut sind und auch für weniger komplexe Konfigurationen ökonomisch sinnvolle Lösungen ermöglichen, stellt das PXI-Format für den Einsatz der LIN 2.0 Interfacebaugruppen die wichtigste Hardwareplattform dar. Nicht zu vernachlässigen sind aber nach wie vor speziell in den Entwicklungsbereichen verbreitete Stand-Alone-Testumgebungen, mit denen sehr häufig nur ein eingeschränkter Funktionsumfang des Steuergerätes geprüft wird, beispielsweise unter Ausnutzung der Diagnosefähigkeit. Für derartige Applikationen sind PC-basierende Systeme am weitesten verbreitet, deren Testressourcen in Form von PCI-Einsteckkarten konfiguriert werden oder als Stand-Alone-Gerät über eine Standardschnittstelle angebunden werden.



Bild 2: LIN 2.0 Interfacebaugruppen für unterschiedliche Hardwareplattformen

Die Notwendigkeit, unterschiedliche Hardwareplattformen zu unterstützen, beeinflusst die Hardwarearchitektur der Schnittstellenbaugruppen maßgeblich. In Abbildung 3 ist das Blockschaltbild der Hardware dargestellt. Man erkennt an der Gegenüberstellung der Architekturen von PCI- und PXI-Boards, dass der Prozessorkern auf Basis eines 32-bit Tricore Prozessors für jede konstruktive Ausführung der Baugruppen identisch ist. Er bestimmt mit seiner Performance und Speicheraufrüstung die Leistungsfähigkeit der gesamten Schnittstellenbaugruppe. Die Anbindung an das austauschbare Host-Interface erfolgt generell über Dual Ported RAM, um eine für alle Interfacetypen einheitlich verwendbare Schnittstelle zu haben, deren erzielbarer Datendurchsatz die Performance der Karte nicht beeinträchtigt.

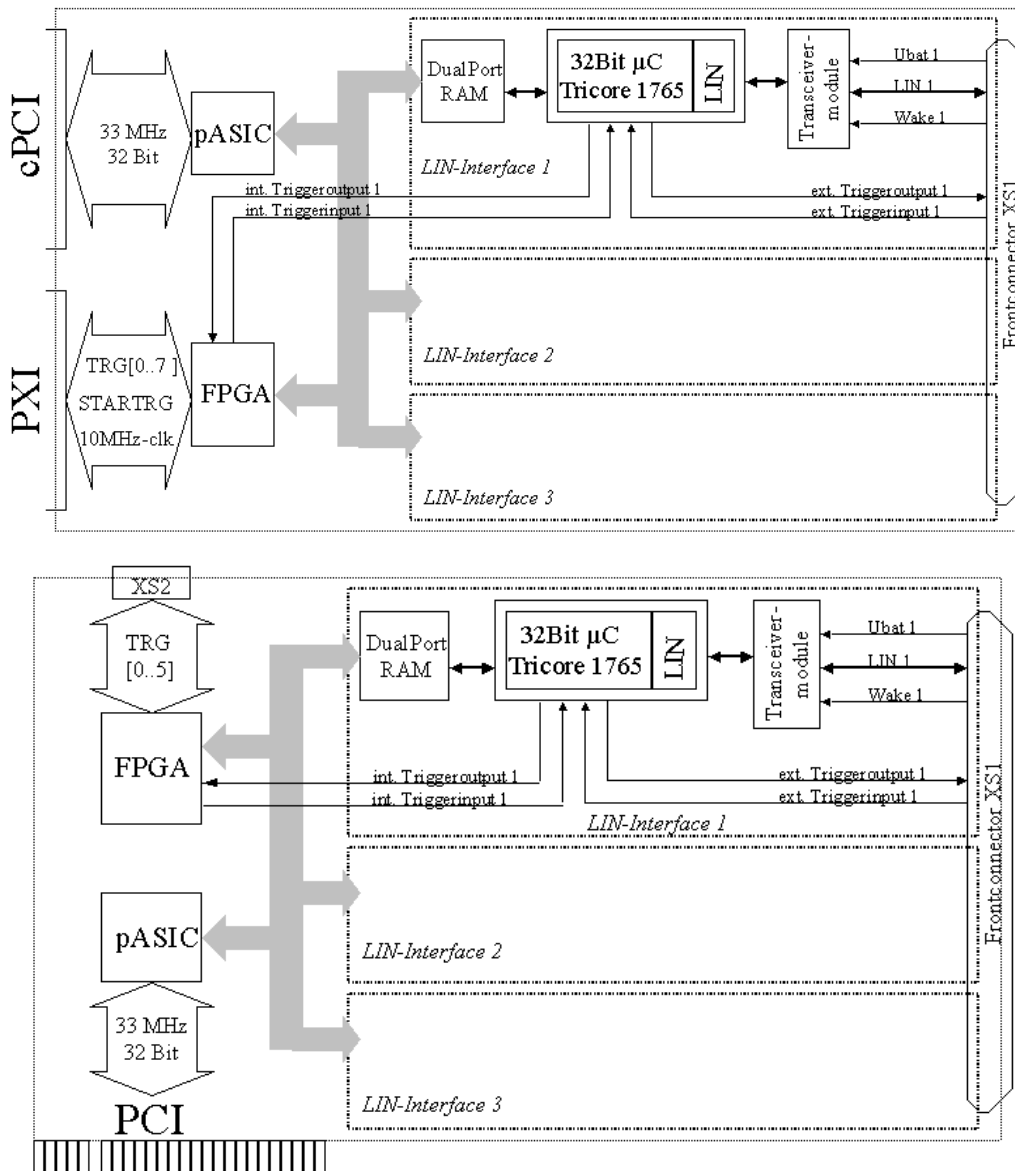


Bild 3: Hardwarearchitektur der LIN 2.0 Interfacebaugruppen in PCI und PXI Version

Die LIN-Interfaces der Baugruppen sind mit steckbaren Transceivermodulen bestückt. Da der LIN 2.0 Standard keinen spezifischen Transceivertyp, sondern nur bestimmte physikalische Eigenschaften vorschreibt, kann die Bestückung der Transceiverbausteine auf diese Weise dem zu prüfenden Steuergerät bzw. LIN-Netzwerk angepasst werden. Pro Interfacebaugruppe sind drei voneinander unabhängige, jeweils mit eigenem Prozessor hinterlegte LIN Interfaces verfügbar, die als Master oder Slave konfiguriert werden können. Damit ist der Aufbau von LIN Clustern gemäß Spezifikation 2.0 möglich.

Für die Integration der LIN 2.0 Interfaces in komplexe Systeme mit weiteren Mess- und Stimulierungsressourcen sind die im Blockschaltbild erkennbaren Triggerleitungen eminent wichtig. Mit Ihrer Hilfe können Systemkomponenten wie z.B. Oszilloskope auf bestimmte LIN-Botschaften getriggert werden.

Softwareunterstützung

Prinzipiell gibt es für den Anwender keinen Unterschied in der Programmierung von PXI- oder PCI-Karten. Das ist wichtig, um die Übertragung von Anwenderprogrammen zwischen unterschiedlichen Prüfständen zu erleichtern, beispielsweise die Nutzung im Entwicklungsbereich am Laboraufbau erstellter Prüfroutinen für den End-Of-Line Test in der Fertigung. Alle plattformspezifischen Daten-Handlings sind in der universellen Treibersoftware abgefangen, die Funktionsaufrufe gestalten sich nach außen hin identisch. Die LIN 2.0 Interfacebaugruppen können über DLL-Funktionen programmiert werden. Bei Installation der im Kartenumfang enthaltenen GÖPEL System Treiber lassen sich die Karten direkt aus Hochsprachen wie VisualC++ oder CVI ansprechen. Dazu stehen folgende Funktionsaufrufe zur Verfügung:

- Abfrage Treiberstatus
- DPRAM Schreiben (Befehl senden)
- DPRAM Lesen (empfangene Botschaft einlesen)
- FPGA File Download

Für Nutzer der graphischen Programmiersprache LabVIEW stehen VI's zum direkten Aufruf der genannten Funktionen zur Verfügung.

Den höchstmöglichen Bedienkomfort sowohl bei der Programmierung der beschriebenen DLL-Funktionen als auch für die Erzeugung von Prüfabläufen unter Ausnutzung des LIN 2.0 Interface Befehlssatzes bietet der als Add-On Tool zur Hardware lieferbare Testsequenzer. Der Testsequenzer ermöglicht die Erstellung kompletter Anwenderprogramme durch Auswahl und Parametrierung vorgefertigter Einzeltestschritte, die aus einer Makrobibliothek abgerufen und zu einem kompletten Prüfprogramm zusammengestellt werden können. Jedes Makro verfügt über eine graphische Eingabeoberfläche, mit deren Hilfe die Anpassung an die spezifische Messaufgabe vorgenommen wird. Abbildung 4 zeigt Beispiele solcher Oberflächen für die Parametrierung von LIN Operationen. Auf der Bedienerbene ist auch die Anbindung von LIN Description Files (ldf) realisiert, mit deren Hilfe sich Botschaftsparameter und Botschaftsinhalt der LIN-Kommunikation direkt aus der dazugehörigen Datenbasis generieren bzw. entschlüsseln lassen.

Die drei LIN-Controller auf den Boards besitzen für Firmware Updates und den Download flüchtiger Programme in den RAM einen Bootloader, der nach dem Einschalten der Versorgungsspannung aktiv ist. Um vom Bootloader-Modus in den Normalbetrieb zu wechseln, ist ein Softwarereset an den entsprechenden Controller zu senden. Nach dem Rücksetzen ist die Slave-Task aktiviert. Die Slave-Task arbeitet mit einer automatischen Baudratenerkennung und dem klassischen Checksummen-Modell, so dass der LIN-Bus ohne Bekanntmachung der Baudrate mit dem LIN-Bus Monitor beobachtet werden kann.

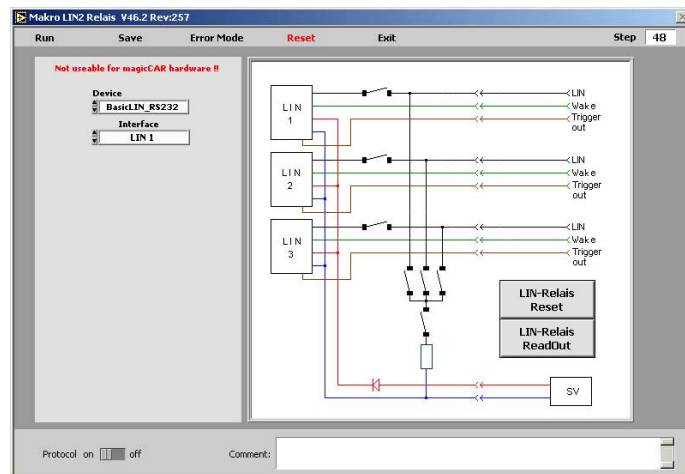
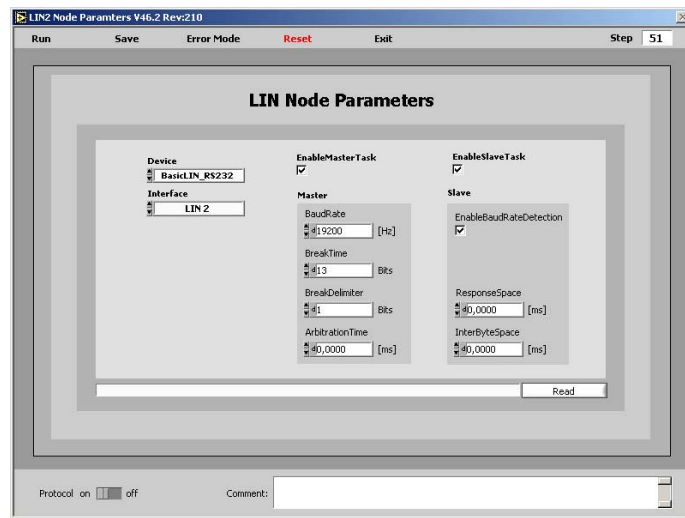
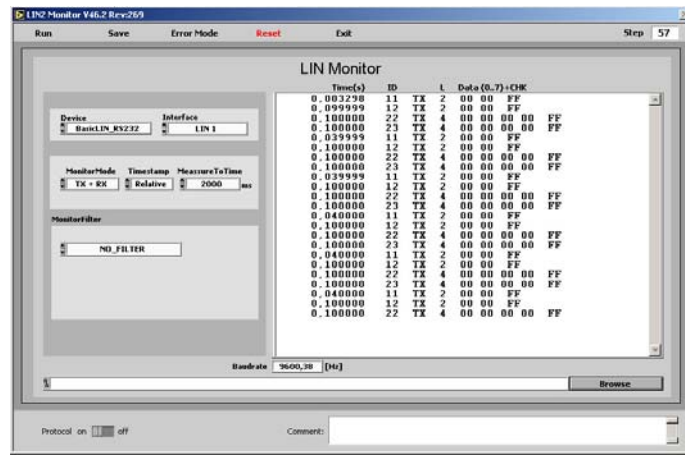


Bild 4: LIN-Bus Monitor und Parametrieroberflächen für LIN-Operationen

LIN 2.0 Befehlssatz und zusätzliche Funktionen

Für den Datenaustausch zwischen Host und LIN-Controller ist eine Befehlsquittierung verfügbar, die durch Setzen des entsprechenden Flags im Botschaftsheader aktiviert werden kann. Es empfiehlt sich, die Befehlsquittierung bei der Erstellung von Anwenderprogrammen möglichst immer zu nutzen. Anderenfalls besteht die Gefahr, bei zeit- bzw. rechenintensiven

Prozessen auf dem Controller, wie z.B. der Tabellenabarbeitung oder der Ausgabe von Rampen, noch nicht abgearbeitete Daten im RAM vom Host aus zu überschreiben.

Neben den bereits beschriebenen Initialisierungsfunktionen sind folgende wesentliche Komplexe von LIN-Operationen ausführbar, die jeweils einen entsprechenden Befehlsvorrat umfassen:

- Abarbeitung von Schedule Tabellen. Insgesamt existieren in der Firmware 16 Schedule Tabellen mit jeweils 256 Einträgen. Das Delay zwischen der Ausgabe einzelner LIN-Botschaften ist als Vielfaches von 25 μ s parametrierbar. Die Option, den Protected Identifier entsprechend der Spezifikation von der Firmware berechnen zu lassen, kann abgeschaltet werden, dadurch ist für Testzwecke das Senden ungültiger Identifier möglich.
- Ausfüllen einer LIN-Botschafts Response Tabelle
Der LIN-Controller kann auf den Empfang von LIN-Botschaften reagieren, indem er eine entsprechende Antwort aus der Response Tabelle ausliest und zurücksendet. Damit entspricht die Response Time den realen Verhältnissen bei der Steuergerätekommunikation im Fahrzeug. Für Testzwecke kann die Ausgabe einer Response unterbunden werden, um das Verhalten von Steuergeräten im Fehlerfall zu untersuchen.
- Steuerung des Sleep/Wakeup-Zustandes von Busteilnehmern
- Definition von Rampen
Der LIN-Controller generiert selbstständig die zyklische Ausgabe von LIN-Botschaften in Rampenform mit Dreieck- oder Sägezahncharakteristik. Minimalwert, Maximalwert, Inkrement sowie Startrichtung der Rampe (aufwärts, abwärts) sind parametrierbar. Die Rampe kann unendlich oder mit einer vordefinierten Anzahl von Zyklen abgearbeitet werden.
- Erzeugen vorübergehender Störungen
Der Befehl gibt dem Anwender die Möglichkeit, das Verhalten von Steuergeräten im Fehlerfall zu testen, indem er abweichend vom normalen Busverkehr fehlerhafte bzw. verstümmelte Responses generiert. Nach Abarbeitung der parametrierten Anzahl fehlerhafter Botschaften werden wieder die ursprünglichen Daten in der ursprünglichen Datenlänge gesendet.
- Modifikation von Timing-Parametern des LIN-Bus wie Baudrate, Arbitrierungszeit Response Space oder dgl. Die Parameter können einzeln gesetzt werden, ohne die kompletten Eigenschaften der ausgewählten LIN-Schnittstelle neu konfigurieren zu müssen.
- Busmonitor mit Empfangsfilter und Time-Stamp-Funktion
- Konfigurieren der Triggerausgänge und Setzen der Relais, welche die 3 LIN-Knoten der Karte auf den Prüfling bzw. zu einem kompletten Netzwerk verschalten. Der Relaisstatus kann über einen entsprechenden Befehl abgefragt werden.
- Diagnose-Konfiguration. Der Anwender kann den Diagnostyp dahingehend auswählen, ob er eine LIN 2.0 spezifikationsgemäße Diagnose durchführen möchte oder ob er die Übertragung von Rohdaten zur Realisierung eigener Protokolle vornehmen will.

Zusammenfassung

Die beschriebenen PXI- und PCI-Baugruppen zur Unterstützung des LIN 2.0 Standards wurden in enger Zusammenarbeit mit Anwendern aus dem Automobilbau bzw. dessen Zulieferbereich entwickelt. Aus dieser Zusammenarbeit resultierten zahlreiche Forderungen und Anregungen bezüglich des Funktionsumfangs und der Bedienbarkeit der Baugruppen, die in der Firmware der LIN-Controller umgesetzt wurden. Nach den bei GÖPEL electronic entwickelten, in vergleichbarer Systemarchitektur aufgebauten PXI und PCI CAN-Baugruppen erwies sich einmal mehr die Richtigkeit des Weges, Interfacebaugruppen mit eigener Intelligenz auszustatten und diese für zeitkritische oder komplexe On-Board-Funktionen zu nutzen. Aufgrund der hohen Aktualität des LIN 2.0 Standards in der Automobilelektronik haben die Baugruppen innerhalb kurzer Zeit weltweite Verbreitung gefunden.

Die Akzeptanz und der Erfolg der PXI/PCI LIN 2.0 Interfaces von GÖPEL electronic ist aber auch ein hervorzuhebendes Beispiel für eine zeitgemäße und sinnvolle Zusammenarbeit im Rahmen des NI Alliance Programmes. Über ein Jahrzehnt hinweg unterteilte man Anfragen nach Komplettlösungen für Mess- und Prüfsysteme in einen universellen und einen anwenderspezifischen Teil, dementsprechend wurden die Zuständigkeiten zwischen Komponentenhersteller und Alliance Partner abgeleitet. Dieses klassische Modell, bei dem sich Systemintegration auf die Verwendung vorgegebener Hardwaremodule und das Schreiben von Bedienersoftware unter LabVIEW oder CVI beschränkt, verfehlt heute die Anforderungen der Praxis und ist überholt. Grund dafür ist die sehr hohe Branchenspezifität der Lösungen mit einem speziell in der Automobilelektronik äußerst kurzlebigen Anforderungsprofil. Es ist zeitgemäßer, eine Dreiteilung des Leistungsumfangs vorzunehmen, indem man zwischen universelle Produktplattform und anwenderspezifische Umsetzung noch eine branchenspezifische Komponente einfügt. Diese Lücke lässt sich im reinen Projektgeschäft aufgrund der vorgegebenen Zeit- und Kostenrahmen nicht schließen. Die hohe Anzahl zu berücksichtigender Spezialgebiete mit teilweise sogar regionalen Unterschieden in der Branchenspezifität (man denke beispielsweise an die unterschiedliche Verbreitung von Busstandards im Fahrzeug oder die abweichenden Diagnoseprotokolle zwischen Europa und den USA) macht es auch für den Plattform-Anbieter nahezu unmöglich, diesen Bedarf abzudecken. Mit der Entwicklung und Vermarktung seiner Automotive Interface Baugruppen ist es GÖPEL electronic gelungen, das Spektrum der National Instruments Plattformen und Komponenten sinnvoll zu ergänzen, um für die aktuellen Testaufgabenstellungen der Steuergeräteprüfung und der Fahrzeugvernetzung Lösungen anbieten zu können.